

FIG. 1

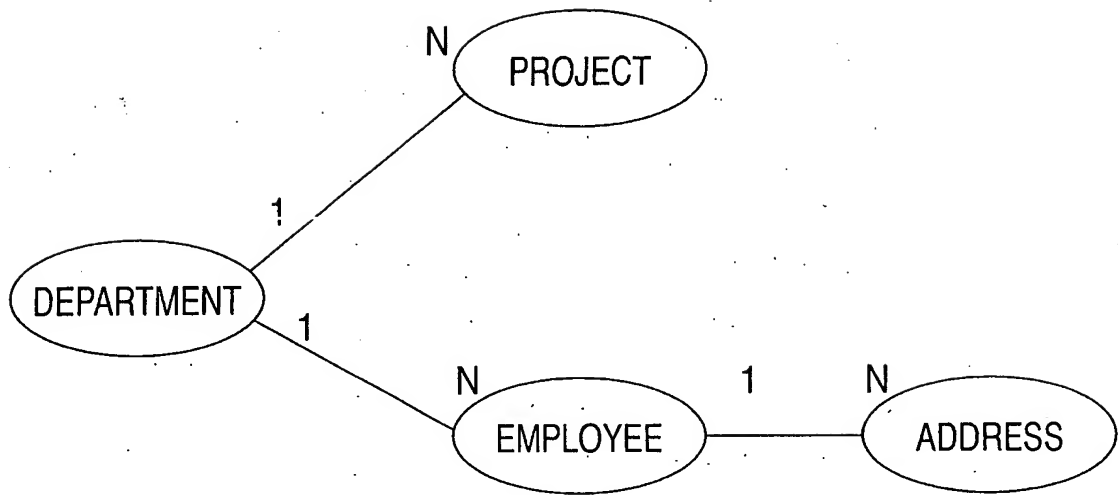


FIG. 2



1. wsPessimisticUpdate
  - a. access-type = pessimisticUpdate
  - b. collection-access = random
  - c. readAhead = null
2. wsPessimisticRead
  - a. access-type = pessimisticRead
  - b. collection-access = random
  - c. readAhead = null
3. wsOptimisticUpdate
  - a. access-type = optimisticUpdate
  - b. collection-access = random
  - c. readAhead = null
4. wsOptimisticRead
  - a. access-type = optimisticRead
  - b. collection-access = random
  - c. readAhead = null
5. wsPessimisticUpdateNoCollisions
  - a. access-type = pessimisticUpdate
  - b. pessimisticUpdateHint = noCollisions
  - c. collection-access = random
  - d. readAhead = null
6. wsPessimisticUpdateExclusive
  - a. access-type = pessimisticUpdate
  - b. pessimisticUpdateHint = exclusive
  - c. collection-access = random
  - d. readAhead = null
7. wsPessimisticUpdatePromotable
  - a. access-type = pessimisticUpdate
  - b. pessimisticUpdateHint = promote
  - c. collection-access = random
  - d. readAhead = null

FIG. 3



ACCESS INTENT POLICY	SELECT SYNTAX	JDBC ISOLATION LEVEL	LOCK TYPE
wsPessimisticUpdate	SELECT <columns> WHERE <primary key column = primary key> FOR UPDATE	REPEATABLE_READ	Update
wsPessimisticRead	SELECT <columns> WHERE <primary key column = primary key>	READ_COMMITTED	Shared
wsOptimisticUpdate	SELECT <columns> WHERE <primary key column = primary key>	READ_COMMITTED	Shared
wsOptimisticRead	SELECT <columns> WHERE <primary key column = primary key>	READ_COMMITTED	Shared
wsPessimisticUpdate NoCollisions	SELECT <columns> WHERE <primary key column = primary key>	READ_COMMITTED	Shared
wsPessimisticUpdate Exclusive	SELECT <columns> WHERE <primary key column = primary key> FOR UPDATE	SERIALIZABLE	Exclusive
wsPessimisticUpdate Promotable	SELECT <columns> WHERE <primary key column = primary key>	REPEATABLE_READ	Shared

FIG. 4



ACCESS INTENT POLICY	JDBC ISOLATION LEVEL	LOCK DISPOSITION
wsPessimisticUpdate	REPEATABLE_READ	Update lock held until end of transaction
wsPessimisticRead	READ_COMMITTED	Shared lock released when cursor closed
wsOptimisticUpdate	READ_COMMITTED	Shared lock released when cursor closed
wsOptimisticRead	READ_COMMITTED	Shared lock released when cursor closed
wsPessimisticUpdate NoCollisions	READ_COMMITTED	Shared lock released when cursor closed
wsPessimisticUpdate Exclusive	SERIALIZABLE	Exclusive lock held until end of transaction
wsPessimisticUpdate Promotable	REPEATABLE_READ	Shared lock released until end of transaction

FIG. 5



ACCESS INTENT POLICY	SELECT SYNTAX	JDBC ISOLATION LEVEL	LOCK TYPE
wsPessimisticUpdate	UPDATE <columns> WHERE <primary key column = primary key>	REPEATABLE_READ	Update promoted to Exclusive
wsPessimisticRead	N/A	N/A	N/A
wsOptimisticUpdate	SELECT <columns> WHERE <primary key column = primary key> AND <over- qualified query predicate>	READ_COMMITTED	Exclusive obtained
wsOptimisticRead	N/A	N/A	N/A
wsPessimisticUpdate NoCollisions	UPDATE <columns> WHERE <primary key column = primary key>	READ_COMMITTED	Exclusive obtained
wsPessimisticUpdate Exclusive	UPDATE <columns> WHERE <primary key column = primary key>	SERIALIZABLE	Exclusive already held
wsPessimisticUpdate Promotable	UPDATE <columns> WHERE <primary key column = primary key>	REPEATABLE_READ	Shared promoted to Exclusive

FIG. 6

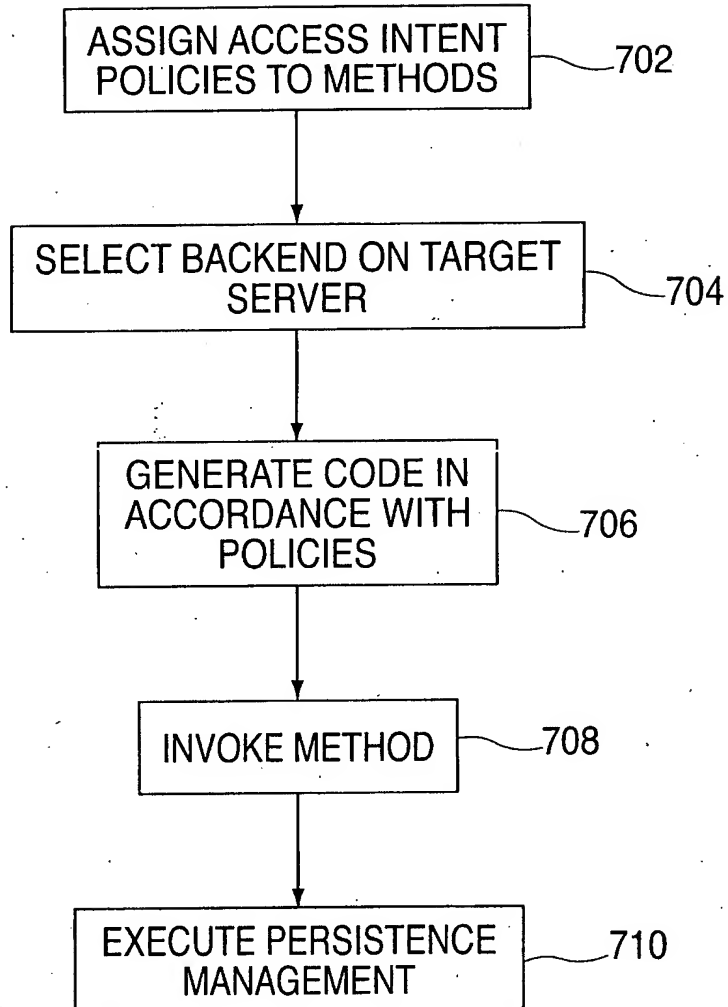


FIG. 7